

# 2017년 프로그래밍언어 개론

-05-

제출일자	2017.04.18.
이 름	정 윤 수
학 번	201302482
분 반	03

## def \_\_str\_\_(self):

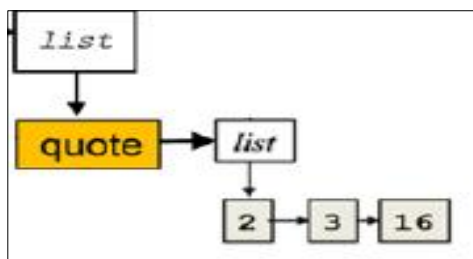
```
def __str__(self):
    result = ''

    if self.type is TokenType.ID:
        result = '[' + NODETYPE_NAMES[self.type] + ':' + self.value + ']'
    elif self.type is TokenType.INT:
        result = '[' + NODETYPE_NAMES[self.type] + ':' + self.value + ']'
    elif self.type is TokenType.LIST:
        result = '(' + str(self.value) + ')'
    else:
        result = '[' + NODETYPE_NAMES[self.type] + ']'

    if self.value is not None and self.type == TokenType.LIST and self.value.type == TokenType.QUOTE:
        result = '[' + str(self.value.next)
    elif self.type == TokenType.LIST and self.value is None:
        result = '()'
    if self.next is None:
        return result
    else:
        return result + ' ' + str(self.next)
```

### 문제 해결 방법

- 저번 과제에서의 출력은 TokenType이 LIST일 일 때 괄호 안에 현재 Node의 value들을 넣어주는 방식으로 출력을 하였다. 하지만 이번 과제에서 QUOTE는 자신을 감싸고 있는 괄호를 출력을 하면 안 된다. 그렇기 때문에 현재 타입이 LIST이고 value의 타입이 QUOTE인 Node에서는 괄호를 출력을 하지 않는 방식을 생각을 하였다. result 값에 새로운 문장을 덧 씌워서 기존에 '()'를 없애고 QUOTE의 타입을 가지고 있는 Node의 next Node를 str함수를 이용 하여서 출력을 하면 재귀문이 Node문을 돌아서 문자들을 출력을 해준다. 하지만 위에 있는 if문으로는 현재 Node의 value가 비어있으면 오류를 발생이 된다 그러므로 value가 None이 아니여야 한다는 조건을 추가를 하였고 현재 Node가 List이고 value가 None일 때 괄호 안을 공백으로 만들어 주기 위해서 새로운 조건문을 추가를 하였다.



**def \_create\_node(self, token) :**

```
elif token.type is CuteType.QUOTE:
    return Node(TokenType.QUOTE)
elif token.type is CuteType.APOSTROPHE:
    node = Node(TokenType.LIST, Node(TokenType.QUOTE))
    node.value.next = self.parse_expr()
    return node
```

( 저번 과제에서 수정을 한 일 부분 만 캡처)

## 문제 해결 방법

- create\_node 함수에서는 각 token의 type에 대응이 되는 CuteType Node의 객체를 생성을 하여 반환을 한다. 이번 과제에서는 APOSTROPHE를 quote의 Node구조의 맞게 작성을 하여 반환을 해야 하는 것을 하여야 하였다. 즉 " ' " 가 입력이 되어지면 ( quote ) 와 같은 역할을 해야 한다. quote의 내부 구조를 구현을 하기 위해서 create\_node함수에서 APOSTROPHE 타입의 token이 if문으로 인하여 판별이 되어지면 quote구조를 생성을 하여 반환을 하여 준다. 먼저 새로운 LIST 타입의 Node를 생성을 하여주고, 새로 만든 Node의 value로 Quote타입의 Node를 생성을 한다. 그 후 만들어진 Quote 타입 Node의 next에는 " ' " 뒤에 오는 내용을 저장을 하면 된다. 모든 과정이 끝나면 APOSTROPHE를 이용을 하여서 quote구조를 만드는 것이 완성이 되어진다. 그 후 만들어진 Node를 반환을 하여 준다.

## 느 낀 점

- 이번 과제에서는 APOSTROPHE를 이용을 하여 quote 구조를 만들기 위해서 Node를 이용을 한 조작을 여러 가지 실험을 해 본결과 Node에 대한 이해가 늘었고, syntax tree를 구성을 하는 방법에 대해서도 좀 더 생각을 하면서 탐구를 해 볼 수 있는 좋은 기회였다.